

BAB I

PERSYARATAN PRODUK

1.1 Pendahuluan

Perkembangan dunia *multimedia* telah berkembang pesat, salah satunya hal yang berperan didalamnya adalah bahasa pemrograman yang membangun aplikasi *multimedia*. Ada beberapa bahasa pemrograman, salah satunya adalah *Java* yang sudah tidak di ragukan lagi perannya dalam membantu meningkatkan perkembangan dunia Teknologi Informasi (TI) khususnya bidang *multimedia*. *Java* adalah bahasa pemrograman yang dapat berjalan di semua platform tanpa harus mengubah kode sedikitpun dengan syarat pada sistem yang digunakan sudah terdapat *JRE (Java Runtime Environment)*. Hal inilah yang menjadi kekuatan *Java* sebagai bahasa pemrograman *multiplatform write once run anywhere*. Dalam pemrograman *java* pengolahan multimedia dapat ditangani atau diolah dengan sebuah *media framework (JMF)*.

JMF API (Java Media Framework Application Programming Interface) adalah *extension* dari *JSE* yang dikonsentrasikan untuk pemrograman pada *multimedia*. *JMF API* merupakan arsitektur yang menggabungkan protokol dan pemrograman *interface* untuk memasukkan data-data media baik itu *audio* maupun video kedalam aplikasi *multimedia* yang dibangun. Seperti proses merekam (*capturing*), *storing* data media, dan *playback* media [Ros07].

JMF 1.0 API (Java Media Player API) memberikan kemudahan bagi programmer *Java* untuk dapat mengembangkan program *Java* yang dapat melakukan *playback* data secara *time-based*. *Time-base* media adalah mengubah data yang diterima dengan berdasarkan waktu, termasuk didalamnya seperti *audio* dan video klip, *MIDI*, dan animasi. Karakteristik *time-based* media adalah dibutuhkannya waktu untuk mengirimkan dan

memproses media [Ros07]. Tugas *java sound API* lebih berfokus pada pemutaran dan pengambilan suara dengan cara memindahkan byte dari data ke dalam format *audio* dan dikeluarkan melalui sistem.

1.1.1 Tujuan

Melihat perkembangan dunia multimedia semakin cepat, maka tercetuslah sebuah ide/pemikiran untuk pembuatan suatu aplikasi pengolahan *Sound/audio* (**Java Sound dengan Pemanfaatan JMF**) sebagai proyek tugas akhir yang dapat digunakan untuk melakukan pemrosesan file sound secara mudah dan pengguna diharapkan memahami pemrosesan seperti penyimpanan, penangkapan (*capture*) dengan proses *setting encode (mixer)* terlebih dahulu dan pengolahan MIDI.

Adapun tujuan utama aplikasi :

- Aplikasi dapat mengakses (*Add*) *file sound* dan *remove file sound* dari aplikasi.
- Aplikasi dapat memainkan (*start/play*) *file sound* dan menghentikan proses (*stop/pause*).
- Aplikasi dapat melakukan pengulangan proses *start/play* dan pengaturan *volume*.
- Aplikasi dapat melakukan *setting encode (mixer)* sebelum melakukan proses *record*.
- Aplikasi dapat melakukan proses *record (capture)* dari input data audio melalui alat bantu *microphone*.
- Aplikasi dapat memainkan *MIDI* dengan berbagai *instruments* dan melakukan proses *record MIDI*.
- Aplikasi dapat menyimpan (*save*) *file sound*.

1.1.2 Ruang lingkup Proyek

Pada *Java Sound API* menyediakan berbagai fasilitas untuk *file input / output* dan *format translations*. Sehingga aplikasi yang dibangun dapat membaca, menulis, dan menerjemahkan antara berbagai format *file sound*. Terdapat beberapa standar format *file sound*, seperti WAV, AIFF, dan AU. Setiap jenis *file* suara memiliki struktur yang berbeda untuk menyimpan data audio dan untuk menyimpan informasi tentang audio data. Sebuah format *file sound* pada *Java Sound API* direpresentasikan oleh *AudioFileFormat* objek. Yang didalamnya terdapat *AudioFormat* objek untuk menjelaskan format audio data yang tersimpan dalam *file*, dan juga berisi informasi tentang jenis *file* dan panjang data dalam *file* [int08].

Dalam membaca suatu *file sound* diperlukan dua tahapan yang paling utama : yang pertama memperoleh Informasi tentang format *audio* data yang tersimpan dalam *file sound*. Dengan *AudioFileFormat* objek dapat memberikan informasi jenis file, panjang data dalam file tersebut, encoding, urutan byte, jumlah saluran, sampel yang tinggi, dan jumlah bit per sampel. Yang kedua membaca *file audio* data dari *file sound*. *AudioInputStream* objek membaca *file* dan melakukan pengujian untuk menentukan apakah jenis *file* tersebut standar format *file sound* (*support*) *Java Sound API*. Jika *file* tersebut tidak *support* maka *Unsupported AudioFileException* akan ditampilkan.

Java sound dengan pemanfaatan *JMF* ini merupakan aplikasi *desktop* dengan menggunakan *swing* yang mirip dengan media *player* pada umumnya. Merupakan program untuk penyimpanan, penangkapan (*capture*), menjalankan (*play*) dan kontrol sebuah *file sound* dalam bentuk format *AIFF*, *AU*, dan *WAV*.

1.1.3 Definisi, Akronim, dan Singkatan.

Tabel I.1 Definisi, Akronim, dan Singkatan.

NO	Singkatan/akronim	Definisi
1.	<i>JAVA</i>	Bahasa pemrograman yang di kembangkan oleh <i>Sun Microsystems</i> .
2.	<i>JMF</i>	<i>Java Media Framework</i>
3.	<i>JRE</i>	<i>Java Runtime Environment</i> .
4.	<i>JSE</i>	<i>Java Standard Edition</i> .
5.	<i>API</i>	<i>Aplication Programming Interface</i> .
6.	<i>AIFF</i>	<i>Audio Interchange File Format</i> merupakan salah satu bentuk file audio yang digunakan untuk menyimpan data bunyi untuk komputer pribadi dan piranti <i>audio</i> elektronik yang lainnya.
7.	<i>Au file format</i>	suatu bentuk <i>file audio</i> yang sederhana diperkenalkan oleh <i>Sun Microsystems</i> . Bentuk itu umum di sistem <i>NeXT</i> .
8.	<i>WAV</i>	<i>waveform audio format</i> merupakan standar <i>format</i> berkas <i>audio</i> yang dikembangkan oleh <i>Microsoft</i> dan <i>IBM</i> . <i>WAV</i> merupakan varian dari <i>format bitstream RIFF</i> dan mirip dengan <i>format IFF</i> dan <i>AIFF</i> yang digunakan komputer <i>Amiga</i> dan <i>Macintosh</i> .
9.	<i>SRS</i>	<i>Software Requirement Specification</i> .
10.	<i>UML</i>	<i>Unified Modeling Language</i> .
11.	<i>Prototype</i>	Suatu produk (model) pendekatan pada satu dimensi atau lebih.
12.	<i>AudioFileFormat</i>	Merupakan suatu objek dalam bahasa pemograman <i>JAVA</i> .
13.	<i>AudioInputStream</i>	Merupakan suatu objek dari <i>JAVA</i> .

1.1.4 Sistematika Laporan

Dokumen disusun dengan menggunakan pendekatan teori *SRS* yaitu sebuah metode penulisan terstruktur untuk pengembangan sebuah *software*. Dengan demikian, penulisan dokumen semakin terstruktur dan efektif serta melingkupi semua aspek secara detail dan terperinci.

Organisasi dari penulisan dokumen dari awal adalah sebagai berikut :

- **BAB I Persyaratan Produk**

Bagian ini berisi pendahuluan, tujuan pengembangan *software*, ruang lingkup dan penjelasan produk yang dibangun secara umum atau *general* dengan menggunakan sedikit bahasa teknis agar memudahkan pemahaman untuk orang awam dan juga bagian ini harus menyediakan gambaran mengenai keseluruhan dokumen.

- **BAB II Spesifikasi Produk**

Bagian ini merupakan penjelasan dari BAB I yang dijabarkan lebih mendalam dan mendetail. Isi dari bagian ini menggambarkan fitur-fitur apa saja yang akan dibuat di dalam *software*, sehingga semua persyaratan, fungsionalitas dan kemampuan *software* dapat dipaparkan dengan jelas dan diidentifikasi dengan unik.

- **BAB III Desain Perangkat Lunak.**

Bagian ini menjelaskan isi dari desain – desain produk secara lengkap dan menggambarkan pemikiran penulis bagaimana aplikasi akan dibangun dengan memperhatikan beberapa faktor.

Perancangan yang dilakukan mencakup :

- ☐ Perancangan *UML*.
- ☐ Perancangan Antarmuka.

Perancangan tersebut akan dijabarkan lagi lebih mendetil dan terstruktur di bagian ini.

- **BAB IV Pengembangan Sistem.**

Bagian ini menjelaskan tentang bagaimana desain yang telah disusun secara terstruktur dan jelas menjadi sebuah produk yang dapat digunakan.

- **BAB V Testing dan Evaluasi.**

Bagian ini berisikan implementasi dan pengujian terhadap program. Dan juga berisikan penjelasan hasil *testing* dari evaluasi dari pengujian terhadap program (*prototype*).

- **BAB VI Simpulan dan Saran.**

Kesimpulan berisi perbandingan-perbandingan antara perencanaan produk sebelum dibangun dengan produk yang dihasilkan. Apakah produk yang dibangun memenuhi tujuan awal yang diinginkan.

Saran berisi hal-hal apa saja yang dapat dikembangkan untuk memberikan kemampuan lebih kepada produk yang dibangun baik secara teknis maupun secara dokumentasi.

1.2 Gambaran Sistem Keseluruhan

Dalam membangun dan mengembangkan aplikasi ini diperlukan menggunakan IDE tertentu dalam membantu mendesain tampilan GUI. Bantuan yang digunakan yaitu *NetBeans* dengan bahasa pemrograman yang dipakai bahasa pemrograman *Java*. **Aplikasi Sound dengan pemaneaan JMF** ini menggunakan *library* yang terdapat pada *JSE* untuk mempermudah dalam membangun proyek ini. Penulis membuat Aplikasi ini untuk diimplementasikan dalam *desktop* komputer.

Diperlukan beberapa perangkat keras yang dibutuhkan untuk mendukung lancarnya pembuatan aplikasi ini. Adapun perangkat keras yang diperlukan dalam pembuatan aplikasi ini: *Processor Intel Pentium 4*, *memory DDR2 512 Mb* dan *Harddisk 40 Gb*, Standar Sound Card Onbord, dan mikrophone.

1.2.1 Perspektif Produk

Produk yang akan dihasilkan merupakan sebuah aplikasi *desktop* yang dapat digunakan pada Semua *operating system* yang *support* terhadap *java* seperti : *Microsoft Windows, Linux, Mac*. Pada aplikasi **Java Sound dengan Pemanfaatan JMF** ini dibuat sepraktis mungkin (*user friendly*) agar lebih mudah untuk digunakan, disini pengguna dapat melakukan pengeditan (pengolahan) *sound* yang di bagi menjadi tiga kategori pengolahan saound yaitu Memainkan *sound (Playing Sound)*, *MixerCapture sound*, dan *MIDI*. Pengolahan *sound* yang pertama SoundBox, didalamnya terdapat fitur-fitur: *add file sound, remove, loop* dengan format *AIFF, AU, dan WAV*, melakukan *record, play, pause, set mono, set stereo, set Volume, time linear*. melakukan pengaturan *sound* yang akan direcord tingkat seperti *frekuensi*, dan terakhir melakukan *save* dengan *format sound* sama dengan *format sound* awal atau berbeda dengan *format* awal. Memainkan *MIDI* dengan berbagai *Instruments* dan melakukan proses *record* pada *MDI* yag dimainkan terakhir melakukan *save* dengan *format sound* sama dengan *format sound .mid*.

1.2.2 Fungsi Produk

Fungsi dari aplikasi *Java Sound dengan Pemanfaatan JMF* :

- Memberikan kemudahan kepada para penggunanya dalam menggunakan aplikasi ini dengan mendesain aplikasi dengan pendekatan *user friendly*.
- Memberikan kemudahan pada pembuatan sebuah permainan efek suara sangat dibutuhkan untuk membuat game yang kita bangun lebih semakin hidup dan terasa nyata, misalnya ketika objek anda mengalami tabrakan, ledakan bom, suara tembakan, dll.
- Memberikan kemudahan kepada para penggunanya dalam melakukan pengeditan *sound* dengan adanya fitur *editing* yang telah disediakan pada aplikasi ini.

- Memudahkan pengguna dalam melakukan perubahan *format sound* pada saat melakukan *save sound*.

1.2.3 Karakteristik Pengguna

Aplikasi ini ditujukan kepada pengguna yang ingin belajar teknologi pengolahan *sound*. Pengguna tersebut dapat dikategorikan sebagai pengguna produktif usia 15 tahun sampai dengan usia 28 tahun, tapi tidak menutup kemungkinan pengguna lain juga dapat menggunakan aplikasi ini untuk pembuatan efek suara sangat diperlukan oleh game.

1.2.4 Batasan-Batasan

Batasan pada sistem :

- Komputer *desktop* yang digunakan harus yang mendukung jalannya aplikasi *java*.
- Aplikasi ini hanya dapat mengakses *sound* dengan *format MIDI, AIFF, AU, dan WAV*.

Batasan pada perangkat keras :

- *Processor Intel Pentium 4*.
- *Memory DDR2 512 Mb*.
- *Harddisk 40 Gb*.
- *Standar SoundCard*.
- *Microphone*.

Batasan pada perangkat lunak :

- *Windows XP Professional SP2*.
- Bahasa pemrograman: *Java*.
- *NetBeans. 6.1*
- *Jdk 1.6.0*.

1.2.5 Asumsi dan Ketergantungan

- Aplikasi ini dapat di operasikan pada komputer *desktop* yang memenuhi batasan-batasan perangkat keras maupun perangkat lunak.
- *NetBeans. 6.1*
- *Jdk 1.6.0.*
- Pengguna hanya dapat melakukan pengeditan *sound* sesuai dengan fitur yang di sediakan oleh aplikasi ini.